Solving Nonlinear Equations

CE 311 K - Introduction to Computer Methods
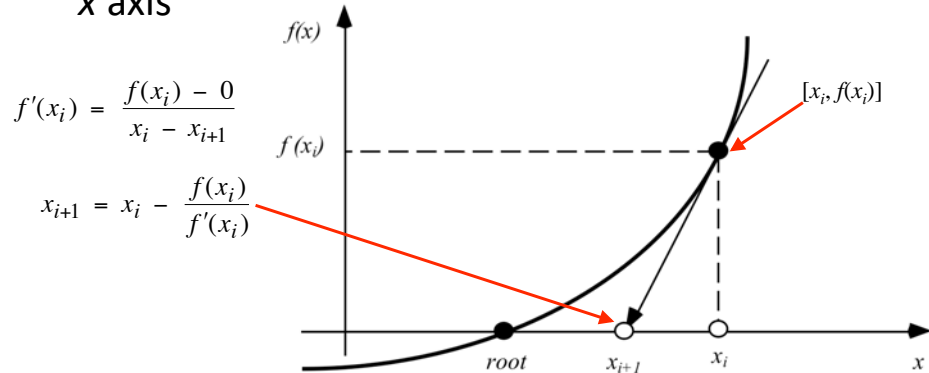
Daene C. McKinney

Solving Nonlinear Equations

- Three Methods
  - Fixed Point Iteration
  - Bisection
  - Newton

Newton's Method

- Extend tangent line from current approximation $[x_i, f(x_i)]$ to where it crosses the x axis

$$f'(x_i) = \frac{f(x_i) - 0}{x_i - x_{i+1}}$$
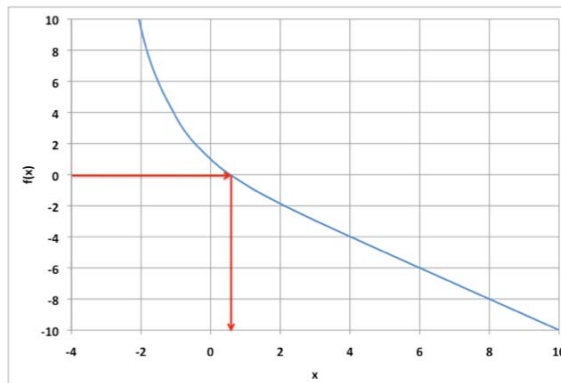
$$x_{i+1} = x_i - \frac{f(x_i)}{f'(x_i)}$$

Example – Newton's Method

$$f(x_i) = e^{-X} - X$$

$$f'(x_i) = -e^{-X} - 1$$

$$x_{i+1} = x_i - \frac{f(x_i)}{f'(x_i)}$$

$$x_{i+1} = x_i - \frac{e^{-x_i} - x_i}{-e^{-x_i} - 1}$$

Example – Newton's Method

```vbnet
Private Sub Button1_Click(ByVal sender As System.Objec
    Dim xn, xo, err, e_max As Double
    Dim i, n As Integer

    n = CInt(TextBox1.Text)
    e_max = CDbl(TextBox2.Text)
    xo = CDbl(TextBox3.Text)
    i = 0
    err = 100

    ListBox1.Items.Add("i" & vbTab & vbTab & _
                       "x" & vbTab & vbTab & "Error")
    Do Until ((i > n) Or (err < e_max))
        xn = xo - f(xo) / dfdx(xo)
        err = Math.Abs((xn - xo) / xn)
        ListBox1.Items.Add(i & "    " & xn & _
                           "    " & err)
        i = i + 1
        xo = xn
    Loop
    ListBox1.Items.Add(" ")
    ListBox1.Items.Add("Root = " & xn)
End Sub

Function f(ByVal x As Double) As Double
    f = Math.Exp(-x) - x
End Function

Function dfdx(ByVal x As Double) As Double
    dfdx = -Math.Exp(-x) - 1
End Function
```

Form1

Maximum # of Iterations `100`

Maximum Error `0.0000001`

Initial Guess `1`

[Go!]   [Stop]

```
i        x                   Error
0   0.53788284273999    0.859140914229523
1   0.566986991405413   0.0513312458779371
2   0.567143285989123   0.00027558218103051
3   0.567143290409784   7.79461013141085E-09

Root = 0.567143290409784
```

3